AMENDMENTS TO THE SPECIFICATION

Please replace paragraph [0004] with the following amended paragraph:

[0004] One way of making content available to Web browsers is for a Web developer to create a static Web page that includes a number of hard-coded computer-readable instructions, such as, for example, in HyperText Markup Language ("HTML"). The static Web page is then stored in a location (e.g., at the server computer system, in a network connectable database, etc.) that is accessible to the Web server. In response to a request from the Web browser, the Web server can transfer the static Web page to the Web browser (e.g., as a result of user input causing the Web browser to access a Uniform Resource Locator ("URL") that corresponds to the static Web page). Upon reception of the static Web page, the Web browser can present the content of the static Web page (e.g., text, graphics, etc.) at the client computer system.

Please replace paragraph [0009] with the following amended paragraph:

[0009] Similar to static Web pages, a dynamic Web page can be cached to provide more efficient access to the dynamic Web page. However, dynamic content (e.g., a price quote) included in a dynamic Web page is typically valid for a limited time (e.g., until values at the location where the dynamic content was retrieved change). Often, dynamic content (e.g., a stock quote) will be valid for a very short period time, for example, only a few seconds. When the value of dynamic content changes, cached dynamic content can become invalid. For example, when a stock price changes from $20.00 to $21.00, a cached value of $20.00 is [[no]] not valid (or even correct). When invalid dynamic content remains in cache, there is some potential for a Web server to provide incorrect dynamic content to a Web browser. For example, when an invalid stock price remains in cache, a Web server might retrieve the invalid stock price from cache instead of re-executing a script to retrieve an updated stock price from a database. Accordingly, cache management mechanisms that attempt to insure the validity of cached dynamic content have been developed.

Please replace paragraph [0016] with the following amended paragraph:

[0016] The foregoing problems with the prior state of the art are overcome by the principles of the present invention, which are directed towards systems, methods, computer program products,

and data structures for registering [[for]] and retrieving database table change information that can be used to invalidate cache entries. A client computer system with a browser is network connectable to server computer system that can generate and provide Web responses (e.g., Web pages and/or Web service results) to the client computer system. The server computer system accesses at least some of the content for Web responses from data tables in a database. The server computer system maintains a cache (e.g., in system memory) that can store provided content so as to increase the efficiency of subsequently providing the same content to satisfy client Web requests.

Please replace paragraph [0026] with the following amended paragraph:

[0026] Figure 1 illustrates an example network architecture that [[facilities]] facilitates configuring cache entries to be dependent on and invalidating cache entries based on changes in database tables in accordance with the principles of the present invention.

Please replace paragraph [0037] with the following amended paragraph:

[0037] In this description and in the following claims, a "Web response" is defined as a human-readable page, such as, for example, a Web page, and/or results from Web service. A Web page can be retrieved from a network, such as, for example, the World Wide Web ("WWW") by accessing a corresponding Uniform Resource Locator ("URL"). A Web page can include one or more HyperText Markup Language ("HTML") instructions that are presentable at a Web browser, such as, for example, browser 102. A Web service provides a mechanism for different applications from different sources to communicate (potentially independent of user interaction) without having to develop custom instructions. Web services can integrate a number of different technologies, such as, for example, Extensible Markup Language ("XML") for tagging data, Simple Object Access Protocol ("SOAP") for transferring data, Web Services Description Language ("WSDL") for describing available services, and Universal Description, Discovery and Integration ("UDDI") for listing available services. Web services can share business logic, data, and processes through a programmatic interface across a network.

Please replace paragraph [0038] with the following amended paragraph:

[0038] In this description and in the following claims, a "Web request" is [[a]] defined as a request for a "Web Response". It may be that a user enters appropriate commands at a user interface to initiate a Web request. For example, a user at client computer system 101 can enter appropriate commands at browser 102 to access a URL corresponding to a Web page. On the other hand, it may be that an application initiates a Web request. For example, a Web services application at client computer system 101 can initiate a Web request to a Web services application at server computer system 111.

Please replace paragraph [0065] with the following amended paragraph:

[0065] The method 200 includes an act of assigning a trigger to the selected data table (act 203). Act 203 can include assigning the trigger to the selected data table when the trigger is not already assigned to the selected data table. For example, a computer system that is network connectable to (or contains) a database can assign a trigger to the selected data table. An assigned trigger can cause versioning information for the selected data table to be updated in the change notification table when content [[a]] in the selected data table is altered. For example, trigger 167 can cause change ID 153 to be updated (e.g., incremented) when content in table 162 is altered (e.g., when a record is inserted into, removed from, or changed in data 163). Similarly, trigger 177 can cause change ID 158 to be updated (e.g., incremented) when content in table 172 is altered (e.g., when a record is inserted into, removed from, or changed in data 173).

Please replace paragraph [0072] with the following amended paragraph:

[0072] A query can be configured to request versioning information for all monitored data tables represented in the change notification table, to request versioning information for monitored data tables with outstanding cache dependencies, or to request versioning information for monitored data tables that have changed (e.g., by including current versioning information in the query). In response to a query, updated versioning information is returned [[form]] from the database to the server computer system. Based on the configuration of the query, the database can return, for example, versioning information for all monitored data tables, versioning information for monitored data tables with outstanding cache dependencies, or versioning information for monitored data tables that have changed.

Please replace paragraph [0092] with the following amended paragraph:

[0092] Line 6 of the first example instructions creates a cache entry (e.g., in cache 112). The cache entry is created with a key of "Pinfo", which can be used to subsequently reference the cache entry. The cache entry is created to store the list of products represented by variable p (e.g., as response content 122) and the cache entry is dependent on the products table in the pubs database represent by variable c (e.g., as table ID 123). The first example instructions may be particularly advantageous when a plurality of Web responses are to include the same portion of content.

Please replace paragraph [0099] with the following amended paragraph:

[0099] For example, it may be that cache entry 121 and cache entry 126 are both dependent on table 162 (e.g., table ID 123 and table ID 128 both correspond to table ID 152). To optimize the efficiency of determining when response content 122 and response content 127 is to be invalidated, a key entry dependent upon table 162 can be inserted into cache 112 (i.e., into system memory). Cache entries 121 and 126 can be made to depend on the inserted key entry. When table 162 is altered, a subsequent query of database 141 can identify that change ID 153 is updated and invalidate [[inserted]] the inserted key entry. Invalidating the inserted key entry automatically invalidates cache entries 121 and 126 rather than requiring both change ID 124 and change ID 129 to be compared directly against change ID 153. Accordingly, the number of database queries and version comparisons can be reduced and network, database, and processing resources conserved.

Please replace paragraph [00107] with the following amended paragraph:

[00107] Likewise, computer system 520 includes serial port interface 546, through which computer system 520 receives data from external sources and/or transmits data to external sources. Serial port interface 546 is coupled to modem 554, through which computer system 520 receives data from and/or transmits data to external sources. Alternately, modem 554 can be a Data Over Cable Service Interface Specification ("DOCSIS") modem or digital subscriber lines ("DSL") modem that is connected to computer system 520 through an appropriate interface. However, as depicted in Figure 5, serial port interface 546 and modem 554 facilitate the

exchange of data with remote computer system 593 via link 552. Link 552 represents a portion of a network, and remote computer system 593 represents a node of the network. For example, remote computer system 593 may be a client computer system that requests Web responses from computer system 520. On the other hand, computer system 593 may be a server computer system that provides Web responses to computer system 520.